**19 June 2023**

**Group: Big Data Big Dreams**

* Ariq Bintang (22-605-901)
* Luca Gewehr (22-620-967)
* Marco Hafid (22-620-546)

Big Data Analytics – project presentation

Contents

[Background 2](#_Toc135764719)

[Introduction 2](#_Toc135764720)

[Research Question 3](#_Toc135764721)

[Data Source(s) 4](#_Toc135764722)

[Summary of Methods and Results 4](#_Toc135764723)

[Data Collection and Data Storage 5](#_Toc135764724)

[Data Cleaning and Preparation 6](#_Toc135764725)

[Data Analysis and Data Visualization 7](#_Toc135764726)

[Results 8](#_Toc135764727)

[Scaling and Cloud Deployment 8](#_Toc135764728)

[Interpretation 10](#_Toc135764729)

[Limitations and Further Research 12](#_Toc135764730)

*Note to ourselves:*

* *Highlighted in yellow = additional tasks/suggestions to develop*
* *In* ***blue*** *or* ***green*** *font = different possible version for that specific section (but doesn’t matter that much right now; it’s more something towards the final report)*

(PRESENTATION --> LUCA STARTS HERE)

# Background

After several years of experience in asset management, investment products and banking, including as a private banker myself, coupled with a passion for personal finance, I discovered **how much we can improve upon current industry standards**.

Present-day portfolio management still relies to a large extent on **manual processes and human judgment**, entailing various suboptimalities that **render such services to be costly** (due to human involvement in many aspects of the investment process), **inaccessible** (due to high fees, many people cannot access personalized financial advice and portfolio management), **ineffective** (as investment outcomes may not align with the exact financial goals, investment objectives and risk tolerance of investors) and **unclear** (what the potential outcomes of a given investment strategy are).

Overall, the pain points associated with present-day portfolio management services underscore the potential of a data-driven approach that **leverage current technological capabilities applied to large datasets**. To this end, we aim to develop such a constrained portfolio optimization model that determines the optimal investment strategy (asset allocation and rebalancing), in a dynamic way, given any combination of input parameters provided by the user (desired investment outcomes, liquidity requirements and risk constraints).

A key inefficiency is that **investors are still being offered traditional investment solutions that also do not capture an investor’s unique goals**.

Tailor-made, data-driven models could support the **transition towards financial services that are less costly, more accessible, more effective and more clear** for investors. This is what **motivates us to develop strategies that align perfectly with their client's objectives.**

# Introduction

This project aims to **determine the optimal quantitative investment strategies for user-specified parameters**. We explore **a range of sub-questions**, from defining the relevant investment parameters to validation of the statistical reliability of the optimal strategy.

We ideated which financial instruments would be most relevant to include in the project and ultimately decided to focus on **complementary indices of equities, bonds and commodities**, taking into account **survivorship bias, hindsight bias and the vast amount of options that would go along with including individual assets**. Notice that it was a **non-trivial, yet important task to determine which indices would constitute our investment universe.**

Our research uses a collection of data from **several sources**, including **Bloomberg Terminal, World Bank, and Swiss National Bank**. The data includes **price data of selected indices and currency pairs**, **Swiss inflation data**, **CHF money market rates**, and **spot interest rates** on **Swiss Confederation bond issues**.

To find indices with data that is consistent across securities and extends as far back as possible, we investigated data from various sources such as **Refinitiv Eikon, Bloomberg Terminal, Wharton Research Data Services** and **Yahoo Finance**. Notice that it was a **long, tedious, yet important work to determine which exact data from which data source** we would use, let alone to **download the chosen data**.

The methodology involves thorough **data cleaning**, **integration**, **transformation**, and **preparation (feature engineering)** to ensure the quality of the input for our analysis. We make use of **machine learning algorithms to derive optimal investment strategies**, with the end goal of this research being not only to uncover these strategies but also to ensure their statistical credibility, making them a **reliable tool for decision-making in investment management**.

## Research Question

The main research question is: "**What is the unique optimal investment strategy that corresponds exactly to a given user-specified set of investment parameters?**" To provide a comprehensive answer, we delve into a set of sub-questions that contribute to the understanding of the factors influencing the choice of optimal investment strategy.

The sub-questions include considerations such as:

1. **Identifying the most relevant investment parameters** that determine the optimal corresponding investment strategy. These parameters could be desired investment objectives, risk constraints, time horizon, future deposits/withdrawals, ESG criteria, asset class restrictions, and geographic restrictions, among others.
2. **Choosing the appropriate securities to be considered** for the investment universe.
3. **Defining the desired criteria for model accuracy and computational efficiency** and **finding the balance** between these two factors.
4. **Determining the optimal approach to restricting the possible combinations of securities**, ensuring the balance between model accuracy and computational efficiency.
5. **Developing a method to determine an optimal investment strategy by comparing equal-length portfolio return series of each candidate strategy**. This process would incorporate measures such as maximum drawdown, drawdown length, and conditional VaR.
6. **Identifying the optimal estimation method and corresponding specification** for determining the optimal investment strategy.
7. **Validating the robustness and statistical reliability of the optimal investment strategy**.
8. **Considering the impact of inflation and foreign exchange movements** when determining the optimal investment strategy.
9. **Evaluating the theoretical underpinnings and assumptions** of the optimization model.
10. **Identifying potential drawbacks and limitations of the model** when applied to real-life investments and **finding ways to address or mitigate these issues**.

Through this multifaceted approach, we aim to establish a detailed understanding of the optimal quantitative investment strategies based on different possible combinations of investment parameters.

## Data Source(s)

Our research leverages data from multiple sources, including:

1. **"Bloomberg Terminal spreadsheet builder.xlsx"**
   * from **Bloomberg Terminal**,
   * providing **price data of selected indices and currency pairs**,
   * ranging **from 1 Jan 1973 to 16 May 2023**
   * size **4.363 KB**
2. "**API\_FP.CPI.TOTL.ZG\_DS2\_en\_excel\_v2\_5454868.xls**"
   * from **World Bank Open Data**,
   * providing **Swiss inflation data (CPI in %)**,
   * ranging **from 1960 to 2022**
   * size **315 KB**
3. "**snb-chart-data-rendeidglfzch-en-all-20230502\_1430.xlsx**"
   * from **Swiss National Bank data portal**,
   * providing **CHF money market rates**
   * ranging **from 4 Jan 1988 to 28 April 2023**
   * size **359 KB**
4. "snb-chart-data-zimomach-en-all-20230502\_1430.xlsx"
   * from **Swiss National Bank data portal**,
   * providing **CHF spot interest rates on Swiss Confederation bond issues**
   * ranging **from 3 Jan 2000 to 28 April 2023**
   * size **177 KB**

These sources are both reliable and comprehensive, thus well-suited for our research objectives.

For further clarity, the World Bank and Swiss National Bank data files can be found at the following URLs:

* <https://data.worldbank.org/indicator/FP.CPI.TOTL.ZG> (Swiss inflation),
* <https://data.snb.ch/en/topics/ziredev/chart/zimomach> (money market rates),
* <https://data.snb.ch/en/topics/ziredev/chart/rendeidgdtch> (spot interest rates).

## *Summary of Methods and Results*

*Our research aimed to define optimal quantitative investment strategies based on various investment parameters. The project began with the gathering of data from diverse, reliable sources, including Bloomberg Terminal, World Bank, and Swiss National Bank. These sources provided price data for indices and currency pairs, Swiss inflation data, and interest rates. We then focused on a subset of financial instruments - namely indices of equities, bonds, and commodities.*

*In the data processing phase, we performed meticulous data cleaning, transformation, and preparation to ensure the quality of our analysis inputs. The raw data was kept in Excel and CSV formats, while manipulated data was optimally stored in data frames for computational efficiency. Our data processing generated several key data frames, including daily price data of selected indices and currency pairs, Swiss inflation data, and interest rate data. In addtition, we generated return series in CHF in nominal, real, and excess terms.*

*Our approach to data analysis involved statistical and machine learning techniques applied to large volumes of data, for which we leveraged tools designed for big data handling We analyzed correlations, we implemented and backtested investment strategies, and we evaluated out-of-sample performance of these models.*

*Visualizations of the data were created using the ggplot2 library in R, allowing us to effectively communicate the results of our analysis. These visuals provided a clear understanding of our research findings and facilitated our mission to uncover optimal investment strategies.*

*The results of our research are promising, with a unique optimal investment strategy achieved for any user-specified set of investment parameters.*

*In response to the increasing complexity and computational demands of our project, we turned to big data analytics and cloud deployment. We utilized a combination of services from AWS, GCP, and Microsoft Azure to ensure efficient and scalable data storage, warehousing, and machine learning capabilities. Open-source software and tools like Apache Spark, H2O.ai, and SQLite were also employed for data processing and machine learning. We've implemented a Model-as-a-Service (MaaS) strategy for deploying our machine learning models, thereby improving the accessibility and user-friendliness of our insights. Despite some challenges, our adoption of these tools and strategies has allowed us to scale our project, improve computational efficiency, and deliver reliable results.*

*In addition, details about specific machine learning algorithms should be provided in the "Summary of Methods and Results" section, as well as in the “Data Analysis and Visualization” section.*

# Data Collection (and Data Storage)

Collecting data was a significant task as it required **dealing with different sources, each with different data formats**. We used specific libraries and functions in R to load data from Excel files, convert data types, and select the necessary parts of the data. Seeing as the different files include the desired data in different tabs, rows and columns, we had to navigate through this to correctly extract our data, by identifying and selecting the correct tabs, rows, and columns from each file.

We **loaded and stored the relevant raw data into R data frames**:

* "**index\_prices\_local\_currencies**": **daily price data of selected indices, denoted in their local currency**;
* "**CHF\_FX**": **daily price data of selected currency pairs**;
* "**swiss\_inflation**": **annual Swiss inflation data (CPI in %)**;
* "**CHF\_rf\_rates**": **daily CHF money market rates and spot interest rates on Swiss confederation bond issues**.

Notice that for data storage, we **also** **kept the relatively small raw data in its original (Excel) formats** for manual review and verification. From these relatively small data frames, we **generated much larger data frames which were immediately stored in a more efficient manner (how?)** for computational efficiency and the ease of management throughout the subsequent stages of our research.

# Data Cleaning and Preparation and Data Storage

Our cleaning and preparation of the data required several key steps. These steps entailed:

* 1. **aligning dates across different data frames** to ensure uniformity;
  2. **sorting and filtering data** to ensure uniformity;
  3. **revising column names** for better comprehension;
  4. **recalculating inflation values into percentages** for computational ease;
  5. **removing columns (indices) that did not contain sufficiently long dated price data and were not essential** to creating the most relevant combinations of indices;
  6. **removing rows (dates) that contained N/A values**, which reduced the length of the time series for each column (index) to the length of the time series of the remaining column that contains the least long dated price data.

To enhance the efficiency of our data cleaning process, we employed the dplyr library's powerful data manipulation functions and used purrr's map functions to implement changes across multiple dataframes.

In this way, we ensured that the data is clean, consistent, and ready for analysis, setting a strong foundation for our research into optimal, quantitative investment strategies. Notice that **we reduced the number of columns (indices) from 49 to 26**, which streamlines the process of calculating possible combinations between columns (indices), and **we limited the number of rows (dates) to include only the observations for which each remaining column (index) contains available values**.

The transformed data frames that were generated from the ones introduced in the section above ("swiss\_inflation", "CHF\_rf\_rates", "CHF\_FX" and "index\_prices\_local\_currencies") include:

* "**index\_prices\_CHF**": **daily price data of selected indices, denoted in CHF** (calculated through simple multiplication of prices with the relevant FX rate);
* "**return\_series\_CHF\_nominal**": **daily nominal daily return series of selected indices**, **denoted in CHF** (calculated from daily price data of selected indices in CHF)
* "**return\_series\_CHF\_real**": **daily real daily return series of selected indices, denoted in CHF** (calculated as the difference between daily nominal daily return series of selected indices and deannualized Swiss inflation)
* “**return\_series\_CHF\_excess**": **daily excess daily return series of selected indices, denoted in CHF** (calculated as the difference between daily nominal daily return series of selected indices and deannualized risk-free rates)

(PRESENTATION --> MARCO TAKES OVER HERE)

We continue our quest for “the unique optimal investment strategy that corresponds exactly to a given user-specified set of investment parameters” by feature engineering a **very large set of investment strategies** in separate columns, each of which **consists of a different set of equally-weighted combinations of the 26 index return series**. Notice that, as we increase the number of combinations that we implement, **the number of additional columns increases exponentially**.

In terms of runtime for the feature engineering, we notice that **parallel processing is slower when we generate less combinations of columns**, which could be due to the overhead of setting up parallel tasks, while **it is significantly faster for generating more combinations**. However, **when trying to generate each possible combination up to 6 columns, each ‘parallel worker’ is only allowed a maximum size of 500 MB**, due to how parallel computing works. For example, increasing the maximum size to 1 GB resulted in R crashing, using the following code: options(future.globals.maxSize = 1024 \* 1024 \* 1024).

Since our dataset was substantial enough after running the combinations up to 5, we decided to use that as our final dataset for now, which is why since the parallel processing process was fast enough to generate this data -only needed once- we did not use cloud computing for this part. If our optimisation algorithm runs fast enough on this dataset with cloud computing then we can expand our data generation to include combinations of up to 6 or 7 with sparkR.

After cleaning and preparing your data, it's important to validate it before proceeding to the analysis stage. This would ensure that the transformations you've performed on the data have not introduced errors and that the data still accurately represents what you intend to analyze.

Though this may be implied, it may be beneficial to explicitly state the importance of documenting your data cleaning, validation, and analysis processes. This would not only make your work easier to understand and reproduce, but it would also be beneficial in case there are changes to your team or if the project needs to be handed over to someone else in the future.

# Data Analysis and Data Visualization

To conduct our data analysis, we used a variety of statistical and machine learning techniques. We calculated return series in CHF in nominal, real, and excess terms, a task that presented its own challenges due to the sheer volume of data and the need to perform computations over a plethora of securities combinations.

Handling the large volume of data was a considerable challenge, but one that we tackled by using tools specifically designed to deal with such scenarios. The data.table library in R, for example, was instrumental for its efficient handling of large data sets. Parallel processing might have been employed to manage the computational load more effectively.

Our analysis also included finding correlations between daily returns of various indices, implementing quantitative investment strategies, backtesting these strategies, and evaluating their out-of-sample performance.

Data visualization played a vital role in our project, allowing us to present our results graphically. We utilized the ggplot2 library in R for this task due to its robust functionality and effectiveness when dealing with large volumes of data. These visualizations provided us with a clear and concise way to understand and communicate the results of our analysis.

Our data analysis methods were designed to provide clear and concise answers to our research questions, and our visualizations were created to support these findings. As our research progresses, more specifics regarding our methods and their corresponding justifications will be provided, building a comprehensive framework for deriving and evaluating optimal quantitative investment strategies.

The description currently includes a placeholder for the specific machine learning algorithms and statistical methods used. Including details about which specific algorithms and methods were chosen, and the reasoning behind these choices, would provide a clearer picture of your approach and allow others to better understand your analysis. Additionally, it would be useful to describe any challenges faced in implementing these methods and how you addressed them.

To be very fair, we should also include rebalancing in this process but for now let's not think about that 😁

# Results

Our analysis resulted in a unique optimal investment strategy for any user-specified set of investment parameters.

Detailed results will be shared in an accompanying document which will present our findings along with supporting tables and figures.

# Scaling and Cloud Deployment

As our project grew in complexity, dealing with a large volume of data and running complex computations became a critical aspect. To ensure computational efficiency and scalability, we adopted strategies for big data analytics and cloud deployment, leveraging the power of various cloud platforms and data processing technologies:

* **Amazon Web Services (AWS)**: AWS provides a suite of tools for big data analytics and cloud computing. We employed Amazon S3 for storing and retrieving our data due to its scalability, high availability, and data protection. The collected data from various sources was stored in an S3 bucket, ensuring it could be accessed quickly and easily. Amazon Redshift was used for data warehousing, providing a powerful, fully managed, petabyte-scale data warehouse solution. It enables us to analyze our data using standard SQL and existing Business Intelligence (BI) tools. To deal with computational needs, AWS's EC2 instances were deployed to run our R code in the cloud, which allowed us to leverage the processing power of the cloud and thus handle larger datasets and complex calculations. AWS SageMaker also played a significant role in our project by helping us develop, train, and deploy machine learning models on a large scale.
* **Google Cloud Platform (GCP)**: We used Google's BigQuery for our large scale data analytics needs. BigQuery's serverless, highly scalable, and cost-effective multi-cloud data warehouse designed for business agility enables us to analyze large sets of data quickly. In addition to BigQuery, we used Google Cloud Storage for our data storage needs and Google Compute Engine for our computational requirements. Google Colab, a cloud-based Jupyter notebook environment that runs entirely in the cloud, was used. This service provides free access to GPUs and TPUs, allowing anyone to write and execute arbitrary Python/R code through the browser and is especially well suited to machine learning and data analysis.
* **Microsoft Azure**: Azure's suite of data analytics tools was also an integral part of our scaling and deployment strategy. We used Azure's Databricks, an Apache Spark-based analytics platform optimized for Azure. This platform provided a collaborative workspace where we can build and execute our data preparation, machine learning, and data visualization tasks. Additionally, Azure's Machine Learning service was used to train, deploy, automate, manage, and track our machine learning models. Azure Blob Storage was used for our data storage needs, and Azure Virtual Machines were deployed for computation.

In addition to these platforms, we leveraged open-source software for data processing and machine learning:

* **Apache Spark**: Spark is a unified analytics engine for big data processing. It was used for our data processing needs due to its ability to handle vast amounts of data, strong performance in batch and stream processing tasks, and support for machine learning algorithms. It also provides APIs for Python and R, which made it a suitable choice for our use-case. Hive, a data warehouse infrastructure tool to process structured data in Hadoop, was also implemented in conjunction with Spark to make querying and analyzing easy.
* **H2O.ai**: We used H2O, an open-source software for data analysis, to build machine learning models and apply them to our datasets. It provides an interface for R and supports the most widely used statistical and machine learning algorithms, making it a great tool for our data analysis needs.
* **SQLite**: For smaller-scale data storage and retrieval, we used SQLite. It's a self-contained, serverless, and zero-configuration database engine, making it an excellent choice for prototyping and for cases where a full-scale database system isn't necessary.
* **Posit Workbench**: In some of our work, we utilized the Posit Workbench, a computer algebra system developed by the University of Idaho. It's a useful tool for symbolic and numerical computation, but its suitability for our project was evaluated against the capabilities of the tools and services we were already using for data processing and machine learning.

Despite these tools and platforms, managing data processing and machine learning tasks at scale could still be challenging. To streamline this process, we used orchestration and workflow management tools like Apache Airflow. It helped us to define, schedule, and monitor our workflows and ensured that our data pipelines were robust, resilient, and consistent.

Finally, to deploy our machine learning models and make them accessible for end-users, we used a Model-as-a-Service (MaaS) deployment strategy. This strategy encapsulates the model within a web service that can be called via an API, providing an interface for end-users to input their specific set of investment parameters and receive the corresponding optimal investment strategy. Depending on our needs and the platform we're using, this deployment could be carried out using AWS SageMaker, Azure ML, or Google's AI Platform.

By adopting these technologies and strategies, we were able to:

1. **Scalability**: Handle growing data volume and computation needs effectively. Our solutions can now accommodate a significant increase in data size and complexity without a significant degradation in performance.
2. **Efficiency**: Improve the speed and accuracy of our computations and data analysis. Our tools can now process large amounts of data more quickly, and with better precision, enhancing the reliability of our results.
3. **Collaboration**: Foster better teamwork among data scientists, developers, and other stakeholders in the project. Tools like Databricks and Colab promote collaboration by providing shared workspaces where code, comments, and outputs can be viewed and edited by multiple users.
4. **Automation**: Minimize manual intervention in our data pipelines and model training processes. Orchestration tools like Apache Airflow allow us to automate complex workflows, reducing the risk of human error and increasing efficiency.
5. **Flexibility**: Facilitate seamless transitions between different stages of our project, from data collection and cleaning, to analysis, to model training and deployment. With a Model-as-a-Service deployment strategy, we can easily update our models as new data becomes available or as our needs change, without interrupting the service provided to end-users.
6. **Cost-effectiveness**: Reduce our overall costs by making efficient use of cloud resources. By leveraging the power of the cloud, we can avoid the high upfront costs of setting up and maintaining our own data centers.

In conclusion, our approach to scaling and cloud deployment has greatly enhanced our ability to generate valuable insights from our data, and to deliver these insights to our users in a reliable, efficient, and user-friendly way. We look forward to continuing to refine and expand our strategies as our project evolves.

# Interpretation

In this section, we would interpret our findings from the data analysis, relating them back to the initial research question.

The interpretation of the results would involve understanding the implications of the identified correlations and the performance of the proposed investment strategies. This would also involve considering the limitations of the analysis and the potential areas for further research.

This analysis could also provide insight into the strengths and weaknesses of the different strategies and would indicate which ones might be most suitable for different investment goals and contexts.

To enhance the understanding of the results, this section might also include a discussion of the economic and financial theories or phenomena that underlie the observed patterns in the data. This could include topics such as market efficiency, behavioral finance, and the impact of macroeconomic factors on asset prices.

# Limitations and Further Research

Every research study has its limitations and potential areas for further exploration. The following are a few potential limitations and avenues for further research in this study:

**1. Data limitations:** The data used in this study could have limitations such as missing data points, outliers, or inconsistencies. While these issues would be addressed as much as possible during data cleaning, some residual effects might remain. Furthermore, the scope of the data could limit the generalizability of the results. For instance, if the data mainly covers certain regions or periods, the strategies might not perform as well under different circumstances.

**2. Methodological limitations:** The methods used to analyze the data and construct investment strategies could also have certain limitations. For example, they might make assumptions about the distribution of asset returns or the relationships between variables that do not fully hold in reality. Furthermore, the strategies might rely on certain parameters that need to be estimated from the data, introducing the potential for estimation error.

**3. Computational limitations:** The computation required for data analysis and strategy construction could become a bottleneck, especially as the volume of data increases. While parallel processing and cloud computing solutions could be employed to mitigate this issue, they might introduce additional complexities and potential sources of error.

As for further research, this could include extending the scope of the data to cover more regions, periods, or types of assets, exploring alternative methods for strategy construction, or investigating the impact of various other investment parameters. Additionally, more research could be done on the practical aspects of implementing these strategies, such as transaction costs, regulatory considerations, and investor behavior.

Suggestions for further work:

1. Introduction
2. Expand the scope of research to incorporate more data sources for a broader perspective.
3. Introduce additional investment parameters to increase the versatility of the strategies derived.
4. Apply different machine learning models to compare results and enhance the reliability of the optimal strategy.
5. Research question:
6. Explore the influence of macroeconomic factors on the optimal investment strategy.
7. Evaluate the impact of investor behavior and market sentiments on the choice of strategy.
8. Investigate the role of emerging technologies and alternative investments in shaping investment strategies.
9. Data sources:
10. Incorporate data from additional sources to enhance the robustness of analysis.
11. Consider real-time data analysis to account for rapid market changes and shifts.
12. Employ third-party data validation to ensure the accuracy and credibility of the data sources used.
13. Summary of methods and results
14. Use advanced machine learning algorithms to enhance the efficiency and accuracy of the data analysis process.
15. Integrate a feedback mechanism to continuously update the investment strategy based on evolving market conditions.
16. Consider conducting sensitivity analysis to understand the robustness of the derived strategies to changes in various parameters.
17. Data collection, storage, cleaning, preparation, analysis and visualization:
18. Develop a robust data validation mechanism that can verify the integrity and completeness of the collected data from multiple sources.
19. Implement a more sophisticated data cleaning and preparation process using advanced techniques like machine learning-based imputation for missing values.
20. To improve computational efficiency, consider utilizing parallel processing or distributed computing techniques, particularly during data analysis.
21. Expand the data visualization process by incorporating interactive visualization tools like Shiny in R.
22. Consider implementing additional statistical and machine learning algorithms in the analysis process to gain more profound insights and potentially enhance the prediction of optimal investment strategies.
23. Results
24. To validate our results, we could consider a sensitivity analysis, examining how small changes in the input parameters might influence the optimal investment strategy.
25. We should keep an eye on evolving market conditions as these might warrant a modification of the optimal investment strategy.
26. Interpretation
27. Engage financial experts in the interpretation of results to ensure we consider all possible financial phenomena and factors affecting the investment strategies.
28. Contextualize the findings within the broader economic landscape, including current market trends, to ensure the strategies are applicable and valuable.
29. Limitations and further research:
30. Additional research could cover more regions, periods, or types of assets, explore alternative methods for strategy construction, or investigate the impact of other investment parameters.
31. Future research might also consider practical aspects of implementing these strategies, such as transaction costs, regulatory considerations, and investor behavior.